
This is the **published version** of the bachelor thesis:

Sarabia Reyes, Arnau; Sanchez Ramos, Carles, dir. TuCanguratge. 2021. (958 Enginyeria Informàtica)

This version is available at <https://ddd.uab.cat/record/248529>

under the terms of the  license

TuCanguraje

Arnau Sarabia Reyes

Resum—Per el projecte final de grau s'ha realitzat un projecte que consisteix en una web, amb l'objectiu de ser intermediària entre usuaris que necessitin un cangur, i un usuari que vulgui ser cangur. Per tant, l'aplicació web, a partir de l'utilització de tecnologies punteres, pretén crear una comunitat d'usuaris, per poder satisfer totes les necessitats derivades del procés de canguratge sense sortir de l'aplicació. L'aplicació web agilitza el procés de creació de peticions, selecció de candidats, així com la comunicació entre famílies i cangurs a través del seu xat intern.

Paraules clau—Canguratge, petició de canguratge, interès, xat, valoració, Cangur, React, Node.

Abstract— For the TFG, a project has been carried out that consists of a website, with the aim of being an intermediary between users who need a babysitter, and a user who wants to be a babysitter. Therefore, the web application, based on the use of cutting-edge technologies, aims to create a community of users, in order to meet all the needs arising from the babysitting process without leaving the application. The web application speeds up the process of creating applications, selecting candidates, as well as communication between families and kangaroos through their internal chat.

Index Terms— babysitting, babysitting request, interest, chat, assessment, babysitter, react, node.



1 INTRODUCCIÓ - CONTEXT DEL TREBALL

A ctualment a conseqüència de la situació pandèmica ocasionada pel COVID-19, s'ha patit un increment d'escoles tancades. Només a Catalunya s'han tancat 5104 escoles durant els mesos de confinament i una mitjana de 560 grups escolars de forma intermitent per positius fins al dia d'avui[1]. Això ha fet que 53.024 alumnes d'entre 3 i 16 anys haguessin de quedar-se a casa amb la conseqüència que molts pares han tingut moltes dificultats a l'hora de deixar els seus fills a càrrec d'una persona especialitzada i idònia per al seu fill[2].

A conseqüència d'aquest increment del tancament de les escoles hi ha hagut un increment en la demanda de canguratge[2]. En algunes situacions, el canguratge s'ha pogut dur a terme per membres de la família a conseqüència del teletreball. Tot i això, moltes famílies s'han vist obligades a recórrer a terceres persones per cuidar dels seus fills

Per poder resoldre aquest problema s'ha creat una web, la qual permetrà des de qualsevol lloc i amb molta facilitat trobar la persona idònia per poder cuidar al teu fill.

La web constarà en la publicació d'una petició de cuidador en el panell, on es podrà indicar les característiques que busquen, els quals permetran als pares buscar la persona que més s'adapti al fill segons les seves preferències. A més, les persones interessades a cuidar dels infants hauran d'especificar algunes dades personals perquè els pares facin primer una criba abans de contactar.

Un cop els pares tenen els candidats, podran contactar amb ells a través d'un xat, amb l'objectiu d'aprofundir una mica més.

Per tant, en aquest article es pot trobar, la solució del problema que s'ha presentat i el procés de com s'ha resolt, desde els objectius, les eines utilitzades, les metodologies, l'arquitectura i com s'ha validat aquest software a través del testing.

-
- E-mail de contacte: arnau_sarabia@hotmail.com
 - Menció realitzada: Enginyeria de Software
 - Treball tutoritzat per: Carles Sanchez
 - Curs 2020/21

2 OBJECTIUS

Els objectius d'un projecte són molt importants a l'hora de tenir un control en tot moment de la seva, per tant, l'objectiu principal del projecte, és poder ajudar als pares amb dificultats per fer-se càrrec dels seus fills a buscar cangurs, per conseqüència del tancament d'escoles, a través d'una web intuïtiva, fàcil d'utilitzar i que es pot utilitzar gratuïtament.

Per assolir aquest propòsit general, el dividirem en diversos objectius. Per tant, s'han determinat els objectius principals, que són els objectius que han d'estar obligatòriament al final del projecte i, els objectius secundaris, que són els objectius que desitjablement haurien d'estar.

2.1 PRINCIPALS

Aquests objectius són els que es van plantejar com obligatoris per poder tenir el projecte en producció.

- **Gestió Usuaris:** Els usuaris podran adquirir dos rols dins de l'aplicació, ja que podran o crear noves peticions perquè els hi cuidin els seus nens o podran sol·licitar realitzar el canguratge d'algun nen. A més, els usuaris podran modificar el seu perfil, filtrar ofertes, interactuar entre ells i fer valoracions entre ells. Finalment, els usuaris podran iniciar sessió i registrar-se amb la finalitat de poder o crear o interactuar amb peticions i amb usuaris, i cada usuari registrat tindrà un perfil d'usuari on especificarà la seva informació i les seves competències professionals en el cas que aquest vulgui accedir al sistema com cangur.
- **Gestió Interessos i Valoracions:** Els usuaris podran tenir interessos per les peticions dels usuaris i podran valorar entre si per tal de donar una experiència més completa i poder escollir la persona que cuidi del seu fill, d'una forma més elaborada.
- **Xat:** El sistema permetrà que els usuaris tinguin accés a un xat entre dos usuaris, quan l'usuari que crea la petició de canguratge, iniciï el xat des de la pantalla d'interessos.
- **Emmagatzematge de les dades:** Un dels objectius del projecte és realitzar l'emmagatzematge de les dades de forma segura, garantir la persistència de les dades, i realitzar aquest emmagatzemament de la forma més òptima possible.

2.2 SECUNDARIS

Aquests objectius són els que no són imprescindibles pel funcionament del programa, però que poden estar en un futur, per una experiència més completa.

- **App:** El projecte podrà tenir una versió App per a mòbils la qual realitzarà la mateixa funcionalitat que la web però especialitzada en els nostres dispositius mòbils.
- **Notificacions:** El projecte tindrà un sistema de notificacions els quals permetran a l'usuari conèixer a través de notificacions internes de la web/app o a través de correus, notificacions sobre els dies que ha de fer de cangur.
- **Premium:** El sistema permetrà tenir preferència a l'hora de promocionar la teva petició de canguratge, i en el cas que vulguis ser cangur, els teus interessos a peticions seran prioritaris.
- **Envíament d'Email:** El sistema permetrà, en el moment que t'interessi una petició, enviar un mail a la persona que ha creat la petició informant-lo.
- **Pagament:** El sistema permetrà realitzar el pagament al cangur a través de la mateixa App.

3 ESTAT DEL ART

Actualment existeixen diferents apps intermediàries per a cuidar els fills, com per exemple, Cangurs de Guàrdia[6], Sitly[7], mil anuncios[8], linkedIn[9], infojobs[10], etc. A continuació, explicarem en que consisteix cada web.

- **Cangurs de Guàrdia:** Consisteix en una web app que et proporciona un cangur en 45 minuts.
- **Sitly:** Connecta a les famílies amb cuidadors i cuidadores. Establerta en 2012, s'ha convertit en la plataforma més gran i de més ràpid creixement per a serveis de cura de nens a Europa i Amèrica Llatina.
- **Mil Anuncios:** És una plataforma gratuïta que permet comprar i vendre tot tipus de productes de segona mà, o oferir serveis, com per exemple, de canguratge.
- **LinkedIn:** És una xarxa social orientada l'ús empresarial, negocis, llocs de treball, etc. Per tant, LinkedIn permet ofertar i trobar feina, entre altres coses, intercomunicant els usuaris.
- **InfoJobs:** És una borsa d'ocupació privada en línia, especialitzada en el mercat espanyol, italià i brasiler, el qual permet buscar i ofertar feina remotament.

Tot i això, les webs ja existents, tenen diferents problemes:

- No estan especialitzades només en canguratge[8][9][10].
- No donen confiança, ja que no estan especialitzades en canguratge i per consegüent el procés d'elecció del candidat no és exhaustiu [8][9][10].
- Les pàgines especialitzades i que donen confiança, s'ha de pagar per poder utilitzar el servei i contactar amb la part interessada.[6][7][8].
- No tens l'opció de triar l'especialista que més t'agradi.[6].
- Hi ha webs internacionals, les quals no estan especialitzades en la cultura local. [7]

Per tant, el que ofereix TuCanguraje és la generació d'un pla de negoci no basada necessàriament per la contribució mensual de l'usuari, sinó basada en la publicitat i en la qualitat de les dades. Per tant, l'usuari tindrà l'opció de poder cuidar un nen sense la necessitat de pagar. A més, la web està especialitzada en canguratge, el que fa que destaquí sobre molta de la competència.

4 METODOLOGIA

Per al desenvolupament d'aquest projecte s'ha utilitzat la metodologia Scrum, on s'han realitzat iteracions setmanals, anomenades "sprints". En cada sprint s'ha definit la feina a fer i els objectius a assolir. Al final del sprint, s'ha valorat la feina feta per reassignar les tasques no realitzades i fer alguna modificació si cal. Al final de cada setmana, s'han anat realitzant revisions amb el tutor per fer un seguiment i guiar sobre modificacions i millores.

Per realitzar el projecte s'ha utilitzat la metodologia Work Breakdown Structure, que és una eina utilitzada per descompondre analíticament un projecte en parts elementals. El plantejament inicial del meu WBS és una primera fase de planificació, on s'ha acabat de perfilar els mòduls i afegir alguna ocurrència més. Una segona fase de disseny on s'han realitzat els requisits, diagrames, primers casos de tests, disseny de la web, etc. Una tercera fase del projecte ha consistit en la implementació del projecte. I per acabar, s'ha realitzat la fase de validació, on s'ha comprovat que tots els requisits estan implementats i que els tests no donen errors.

5 PLANIFICACIÓ

Per el desenvolupament d'aquest projecte s'han planificat 4 fases ben diferenciades en el diagrama de gantt (veure Anexe 1), que son les següents:

- **Planificació:** En aquesta fase s'ha realitzat totes les tasques relacionades amb la decisió de les eines de treball, realitzar la proposta de treball i en definitiva, decidir com ha de ser el projecte.
- **Disseny:** Aquesta fase ha sigut de les més importants, ja que és on es dissenya la base de com serà el projecte, i és on es dissenya i planteja totes les funcionalitats i forma del projecte.
- **Implementació:** Aquesta fase del projecte és la més llarga, ja que consisteix a realitzar totes les tasques de realització del projecte, a partir de tots els diagrames, mapa de pantalles, base de dades, definits en la fase de disseny.
- **Validació:** La fase de validació consisteix a demostrar i verificar que totes les especificacions definides en la fase de disseny, com els requisits, diagrames de flux, casos de test, s'han complert correctament i tenen el funcionament esperat.

La planificació del projecte ha estat bastant realista, ja que s'han complert totes les fases i tasques en el temps esperat, tot i això, hi ha hagut una tasca no contemplada i que ha requerit un total de 30 hores aproximadament, que ha sigut una formació a través de la plataforma de platzi, ja que els llenguatges nous requerien moltes funcionalitats i llibreries noves que no es van contemplar al principi del projecte.

Aquesta formació ha requerit realitzar algun canvi en l'ordre d'execució de les tasques, com per exemple, en primera opció realitzar el backend, per després poder realitzar la formació de les eines de connexió amb el backend.

6 REQUERIMENTS

Una de les parts més important del projecte és la definició dels requisits, ja que, aquests determinaran l'abast del projecte i la funcionalitat d'aquest. Per tant, els nostres requisits els dividirem en funcionals (que ha de fer el sistema) i no funcionals (com ho ha de fer). Per realitzar aquests requeriments s'ha tingut en compte el diagrama de flux (veure Anexe 2).

6.1 FUNCIONALS

Els requeriments funcionals de la nostra aplicació és la base per poder realitzar l'aplicació, ja que ens dona una idea global de tota la funcionalitat que volem que faci la nostra aplicació, i són els següents:

- **RF1.** El sistema permet als usuaris loggejar-se i registrar-se.
- **RF2.** El sistema permet als usuaris modificar el seu

perfil

- **RF3.** El sistema permet als usuaris crear peticions de canguratge.
- **RF4.** El sistema permet als usuaris filtrar per les peticions de canguratge actuals.
- **RF5.** El sistema permet crear un interès en una petició de canguratge
- **RF6.** El sistema permet valorar un usuari que ha creat una petició de canguratge.
- **RF7.** El sistema permet veure les valoracions d'un usuari que ha creat una petició de canguratge.
- **RF8.** El sistema no ha de permetre crear una petició a un usuari que no s'ha logejat.
- **RF9.** El Sistema no ha de permetre interactuar amb una petició a un usuari que no s'ha logejat.
- **RF10.** El sistema ha de permetre veure els interessos que té un usuari logejat.
- **RF11.** El sistema ha de permetre veure els chats actius d'un usuari logejat.
- **RF12.** El sistema ha de permetre veure si els usuaris estan online o no dins dels chats actius.
- **RF13.** El sistema ha de permetre veure el perfil de l'usuari interessat en la petició.
- **RF14.** El sistema ha de permetre veure les valoracions de l'usuari que està interessat en la petició.
- **RF15.** El sistema ha de permetre valorar un usuari que està interessat en la petició.
- **RF16.** El sistema ha de crear un nou xat quan un usuari accepta un interès.
- **RF17.** El sistema ha de permetre llistar tots els usuaris amb un xat actiu.
- **RF18.** El sistema ha de permetre canviar els missatges del xat entre la llista del xat.
- **RF19.** El sistema ha de permetre tancar la sessió del usuari.
- **RF20.** El sistema ha de permetre avisar si l'usuari d'un xat està en línia o no.

6.2 NO FUNCIONALS

Els requeriments no funcionals són característiques generals del sistema o restriccions a tenir en compte. Poden anar variant al llarg del projecte. A continuació es pot veure un recull de tots els proposats durant el TFG juntament amb l'estat final (realitzat, desestimat).

- **RNF1** El sistema ha de recarregar les peticions cada 5 segons.
- **RNF2** El sistema permet tenir un màxim de 1 petició de canguratge per usuari.
- **RNF3** El sistema ha de recarregar el xat cada segon.
- **RNF4** El sistema ha de recarregar els interessos cada 5 segons.
- **RNF5** El sistema ha de permetre crear una contrasenya entre 1 i 15 caràcters com a màxim.
- **RNF6** El sistema permet tenir una valoració com a màxim d'un usuari A a un usuari B.
- **RNF7** El sistema ha de permetre filtrar per la ciutat de totes les peticions de canguratge.

7 EINES

Pel desenvolupament del projecte s'han valorat diferents frameworks i tecnologies, diferents. El fet d'escollir un Framework o una llibreria, ha sigut molt beneficiós, ja que aporta un codi organitzat des del primer moment, t'estalvia molt temps, t'ajuda en temes de seguretat, etc.[4]

Dins de totes les possibilitats es trobaven tecnologies orientades a web, com Angular, Vue.js, React.js.

- **Angular[11]:** És un framework per aplicacions web desenvolupat en TypeScript, de codi obert, mantingut per Google, que s'utilitza per crear i mantenir aplicacions web d'una sola pàgina.
- **Vue[12]:** Vue.js compta amb una arquitectura d'adaptació gradual que es centra en la representació declarativa i la composició de components. La biblioteca central es centra només en la capa de vista. Les característiques avançades necessàries per a aplicacions complexes com l'enrutament, la gestió d'estats i les eines de construcció s'ofereixen a través de llibreries i paquets de suport mantinguts oficialment, amb Nuxt.js com una de les solucions més populars.
- **React[13]:** És una biblioteca Javascript de codi obert dissenyada per crear interfícies d'usuari

amb l'objectiu de facilitar el desenvolupament d'aplicacions en una sola pàgina. És mantingut per Facebook i la comunitat de programari lliure. En el projecte hi ha més de mil desenvolupadors lliures.

Finalment, s'ha decidit realitzar el projecte amb el front-end de React.js, ja que la seva integració és molt flexible, l'organització del codi és molt intuïtiva i a més, React.js actualment és una de les llibreries més completes i més utilitzades, dins de les empreses grans i petites, com Facebook, Dropbox, Netflix, Instagram, etc.[3]. Per una altra banda, React.js proporciona l'eina de React Native, que permet generar aplicacions mòbils utilitzant el mateix codi JavaScript de l'aplicació web.

A més, la projecció que té React.js en el futur és molt creixent, i de cara a currículum és l'opció que actualment més sortida laboral té [5].

Dins de les eines de react i node s'han utilitzat les següents:

- **Axios:** és una llibreria JavaScript que pot executar-se en el navegador i que ens permet fer senzilles les operacions com a client HTTP, de manera que podem configurar i realitzar sol·licituds a un servidor i rebrem respostes fàcils de processar[14]. Aquesta llibreria s'ha utilitzat per realitzar les peticions a l'API de node.js i poder obtenir la informació necessària del servidor.
- **Props:** El terme "render prop" es refereix a una tècnica per compartir codi entre components en React utilitzant una propietat el valor és una funció. Un component amb una render prop pren una funció que retorna un element de React i ho crida en lloc d'implementar la seva pròpia lògica de representació[15].
- **Hooks:** Els Hooks són una nova API de la llibreria de React que ens permet tenir estat, i altres característiques de React, en els components creats amb una function. Hooks és ganxo i, precisament, el que fan, és que et permeten enganxar els teus components funcionals a totes les característiques que ofereix React.[16]
- **Express:** És un framework per NODE.JS que serveix per ajudar-nos a crear aplicacions web en menys temps ja que ens proporciona funcionalitats com l'enrutament, opcions per gestionar sessions i cookies, entre altres coses.[17]
- **Componentes:** són elements independents i poden ser reutilitzats, a més, descriuen com han de visualitzar-se i com han de comportar-se. [20]

Per al back-end, finalment es va optar per Node.js ja que és la opció que millor s'adapta a react.js. Node.js es un entorn d'execució multiplataforma, de codi obert per

desenvolupar aplicacions web. Aquesta llibreria s'executa sobre JavaScript i ha sigut creada per Google.

Un dels grans avantatges de Node.js és que treballa amb una programació asíncrona, el que permet interactuar amb molta més facilitat entre front-end i back-end.

Un cop decidit el llenguatge i les llibreries a utilitzar com a eina de desenvolupament s'ha utilitzat visual code, ja que és el software que millor integra el back-end i el front-end, i a més permet utilitzar d'una forma àgil tota la inclusió de llibreries externes i git-hub, que ha sigut el control de versions que s'ha utilitzat per poder anar fent un codi segur, versionat, i en el núvol. Tot i això, no s'ha pogut extreure tot el potencial de git al ser un sol desenvolupador.

La part relacionada amb SQL s'ha implementat amb oracle, ja que proporciona un servidor de base de dades SQL (Structured Query Language) molt ràpid, multi-threaded, multi usuari (pensant en un futur) i robust. El servidor oracle està dissenyat per a entorns de producció crítics, amb alta càrrega de treball així com per integrar-se en programari per a ser distribuït.

Per al desenvolupament de la base de dades s'ha utilitzat dos programes.

- **SQL Modeler:** Oracle SQL Developer Data Modeler és una eina gràfica gratuïta que millora la productivitat i simplifica les tasques de modelatge de dades. A l'utilitzar Oracle SQL Developer Data Modeler, els usuaris poden crear, navegar i editar models lògics, relacionals, físics, multidimensionals i de tipus de dades[18].
- **SQL Developer:** Oracle SQL Developer és un entorn de desenvolupament integrat i gratuït que simplifica el desenvolupament i la gestió de la base de dades Oracle tant en els desplegaments tradicionals com en els de Cloud. SQL Developer ofereix un desenvolupament complet de les vostres aplicacions PL / SQL, un full de càlcul per executar consultes i scripts, una consola DBA per gestionar la base de dades, una interfície d'informes, una solució completa de modelització de dades i una plataforma de migració per Bases de dades de tercers a Oracle.[19]

8 DESENVOLUPAMENT

El desenvolupament del projecte s'ha realitzat en dues parts diferenciades, que són el frontend realitzat en react i el backend realitzat en nodejs, que és la API la qual s'utilitza per fer peticions des de el front-end.

Per tant, en aquest apartat s'explicarà en detall l'estructura i el contingut del projecte.

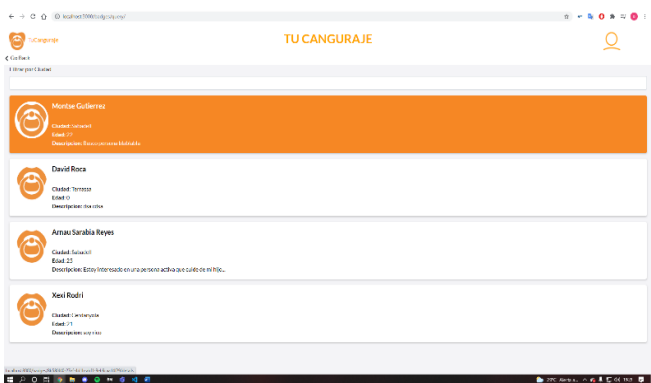
8.1 FRONTEND

L'organització del frontend s'ha realitzat a través d'un router que s'encarrega d'anar distribuint les peticions get i les accions, i d'aquesta manera poder navegar a través de totes les pantalles de la web.

A continuació, explicarem com funciona el flux de pantalles, i les funcionalitats de la web de forma detallada.

En primera instància, hi ha la pantalla principal on s'informa del funcionament de la web i on hi ha informació sobre el projecte. En aquesta pàgina es pot seleccionar la ciutat en la qual volem començar a filtrar per trobar una oferta de canguratge.

Un cop filtrem ens sortirà un llistat amb totes les ofertes d'aquesta ciutat, i podrem seleccionar l'opció que més interressi l'usuari.



Il·lustració 1: Llistat de peticions de canguratge

Quan es selecciona la petició de canguratge, es mostrarà una pantalla amb el detall de la petició i informació addicional, sobre aquesta petició. Per interactuar amb la petició és necessari iniciar sessió a la web o registrar-se en el cas que encara no sigui un usuari del sistema.



Il·lustració 2: Perfil de Petició sense logejar

La part del registre està orientada per tot tipus d'usuari, tot i així, els únics camps obligatoris es marcaran amb '*', amb l'objectiu que, si un usuari vol crear una petició de

canguratge no es necessari que ompli tots els camps del seu perfil (experiència, estudis, etc.).

Un cop creat el perfil, l'usuari ja podrà logejar-se al sistema.

Quan un usuari està logejat realitzarà un tipus d'accions depenent si l'usuari vol crear una petició de canguratge o vol buscar feina.

En el cas que vulgui crear una petició anirà al llistat de peticions i clicarà la opció de "crear petició". Es mostrarà per pantalla un formulari on es carregaran prèviament les dades del perfil i es podrà indicar les dades de la petició (horari, preu hora, descripció). En el cas que l'usuari ja tingui una petició creada es carregaran les dades de la petició per poder actualitzar-la, ja que un usuari no pot tenir dues peticions actives.

En el cas que l'usuari vulgui buscar feina, podrà accedir a la petició de canguratge que més l'interessi i clicar el botó de "Me Interesa".



Il·lustració 3: Petició de canguratge logejat

Un cop realitzat aquest procés es realitzarà una petició a la API i es crearà el registre, controlant si ja hi ha un interès a aquella petició de canguratge.

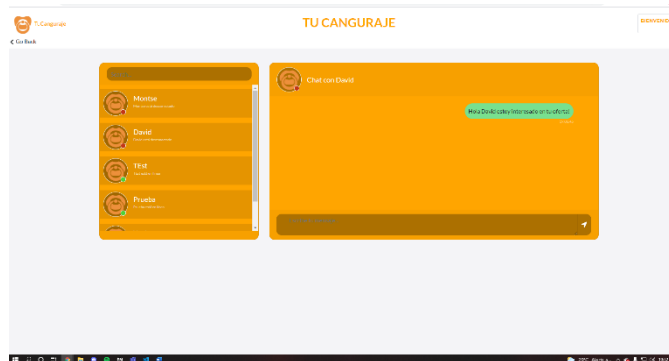
Un altre apartat important dins de la web és la secció de menu de navegació, ja que permetrà a l'usuari veure les opcions que s'explicarà a continuació:

- **Xat:** Aquesta opció permetrà veure tots els xats actius de l'usuari.
- **Interessos:** Aquesta opció permetrà veure tots els interessos sobre l'usuari.
- **Log-out:** Aquesta opció permetrà a l'usuari tancar la sessió.

En el moment que un usuari té interessos per part d'un altre usuari, aquest podrà veure el perfil de l'usuari que està

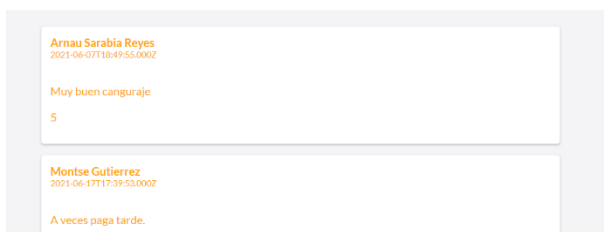
interessat en fer de cangur i en el cas de que encaixi amb les expectatives que té, podrà clicar el botó de “Iniciar Xat”. Aquest botó creara un registre nou de xat per poder comunicar-se amb l’usuari i així poder acabar de saber si és la persona idònia per cuidar al seu fill.

L’usuari que està interessat en fer de cangur no podrà accedir al xat amb la persona de la oferta fins que aquesta no seleccioni la opció de iniciar Xat.



Il·lustració 4: Xat

Per una altra banda, l’usuari que ha realitzat la petició, podrà veure totes les valoracions que té aquest usuari i d’aquesta manera tenir més referències sobre ell. A més, també podrà crear valoracions per ajudar a altres usuaris a tenir una opinió externa sobre l’usuari que far de cangur.



Il·lustració 5: Valoracions

8.2 BACKEND

L’organització del backend s’ha realitzat amb un router igual que el front-end que redirigeix segons la petició a un mòdul o un altre.

El backend està configurat per realitzar la connexió a través d’uns paràmetres de connexió amb la base de dades d’oracle.

Un cop s’inicia el servidor, queda a la espera de qualsevol petició. Un cop rebuda la petició, a través de express, accedirà a la ruta de la classe a la que s’ha realitzat la petició, i s’identificarà el tipus de petició CRUD que s’ha realitzat (Create, Read, Update, Delete). Express proporciona

seguretat a l’hora de realitzar les peticions entre el frontend i l’API del backend.

La petició pot tenir dos tipus de dades, per paràmetre o un json. Per paràmetre únicament es passen dades que no són sensibles, com ID d’usuari, ID de peticions, nom d’usuari, etc. En canvi tot el tema de contrasenyes, DNI, emails, es passaran desde el json.

8.3 BASE DE DADES

Per realitzar la base de dades, en primera instància s’ha realitzat el diagrama E-R a partir dels requeriments obtinguts (veure Anexe 4).

Un cop fet el diagrama s’ha validat, comprovant que la base de dades realitzava correctament la funcionalitat desitjada. Un cop, validat s’ha realitzat el diagrama modeler a través de l’eina d’oracle “data modeler”, el qual et permet d’una forma molt visual realitzar tot el procés de creació de taules, atributs i relació entre taules (PK, FK).

Finalment, un cop obtingut el diagrama modeler (veure Anexe 5), s’ha pogut realitzar una exportació del codi, generat automàticament per poder realitzar les taules amb les relacions entre elles.

Les taules generades han sigut les següents:

- **Usuari:** Aquesta taula registrarà totes les dades dels usuaris sense distingir els diferents rols que hi ha.
- **Petició:** Aquesta taula registrarà totes les peticions de canguratge que hi ha dels usuaris, controlant que només pugui haver una per persona.
- **Interesa:** Aquesta taula registrarà tots els interessos que hi ha d’un usuari A a un usuari B que té una petició activa.
- **Acord:** Aquesta taula registrarà un historial amb totes les relacions que han hagut entre els diferents usuaris
- **Xat:** Aquesta taula registrarà la capçalera de tots els xats disponibles
- **HistorialXat:** Aquesta taula registrarà totes les línies de xat de les diferents capçaleres.
- **Valoracions:** Aquesta taula registrarà la capçalera de totes les valoracions entre usuaris.
- **HistorialValoracions:** Aquesta taula registrarà un històric de totes les valoracions que hi ha hagut.

9 TEST

Una de les parts més important a l'hora de fer un projecte és la validació del mateix, ja que aquest procés, implica garantir el bon funcionament. Per tant, s'ha realitzat tant test per la part de react com per la part de nodejs.

9.1 FRONTEND

La realització del frontend bàsicament s'ha realitzat a través test case, els quals es van definir al principi del projecte i consisteixen en uns fluxos d'execució els quals generen un input en un mòdul concret del programa, i s'ha de comprovar si la resposta és la esperada segons els requisits definits del sistema.

Els units test definits són els següents:

UT1 - Registrar un nou usuari		
1	Input:	Usuari ja existent
	Output:	Email existent
2	Input:	Usuari no existent
	Output:	Anar a pantalla principal

UT2 - Iniciar sessió		
1	Input:	Email inexistent
	Output:	Email inexistent
2	Input:	Contrasenya erronia
	Output:	Contrasenya incorrecta
3	Input:	Login correcte
	Output:	Anar a pantalla principal

UT3 - Crear nova petició d'usuari		
1	Input:	Usuari no logejat
	Output:	No ha de tenir l'opció de crear petició
2	Input:	Usuari logejat amb petició ja creada
	Output:	Contrasenya incorrecta
3	Input:	Pantalla d'editar petició
	Output:	Pantalla de crear petició

UT4 - Veure Valoracions		
1	Input:	Usuari amb valoracions
	Output:	Llistar totes les valoracions
2	Input:	Usuari sense valoracions
	Output:	Mostrar missatge informant que no hi ha valoracions.

UT5 - Crear Valoració		
1	Input:	Usuari no logejat
	Output:	No ha de tenir accés a crear valoració
2	Input:	Usuari logejat sense valoració per aquell usuari
	Output:	Obrir pantalla per crear valoració
3	Input:	Usuari logejat amb valoració per aquell usuari
	Output:	Obrir pantalla d'editar valoració

UT6 - Tancar Sessió		
1	Input:	Usuari no logejat
	Output:	No ha de tenir accés
2	Input:	Usuari logejat
	Output:	Anar a pantalla principal

UT7 - Veure Xats		
1	Input:	Usuari no logejat
	Output:	No ha de tenir accés
2	Input:	Usuari logejat
	Output:	Anar a pantalla de xats.

UT8 – Veure Interessos		
1	Input:	Usuari no logejat
	Output:	No ha de tenir accés
2	Input:	Usuari logejat sense interessos
	Output:	Ha de mostrar un missatge indicant que no té interessos.
3	Input:	Usuari logejat amb interessos
	Output:	Mostrar interessos

UT9 – Iniciar Xat		
1	Input:	Usuari sense xat creat per aquesta persona
	Output:	Crear xat i mostrar pantalla de xat
2	Input:	Usuari amb xat creat per aquesta persona
	Output:	Mostrar pantalla de xat

UT10 – Iniciar Xat		
1	Input:	Usuari no logejat
	Output:	No ha de tenir accés
2	Input:	Usuari logejat modifica valors no bloquejats
	Output:	Anar a pantalla principal.

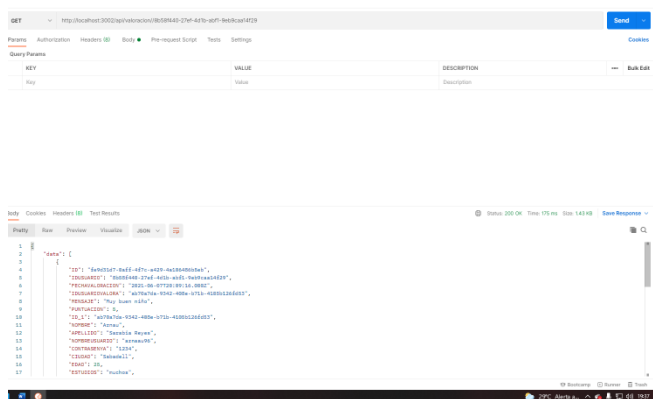
Un cop realitzades aquestes proves de caixa negra em puguem comprovar que el sistema respon seguint els requisits definits al inici del programa i podem validar que el sistema funciona correctament per la part del frontend.

9.2 BACKEND

Per realitzar el unit test de cada petició s'ha utilitzat el software de POSTMAN, el qual consisteix en realitzar una petició manualment recreant el contingut de la mateixa, i comprovar que la resposta de la petició es la esperada.

Per tant, el procediment a seguir a l'hora de realitzar el backend, ha sigut la de crear una petició nova i abans d'integrar-la en el frontend tesetejar-la i d'aquesta manera assegurar-nos que estàvem introduint una funcionalitat que funcionava correctament.

A més, per a cada funcionalitat s'ha fet proves per totes les casuístiques, el que coneixem com statement coverage. Per fer aquestes proves de caixa blanca, es comprova totes les



Il·lustració 6: Exemple de postman

diferentes casuístiques que podem tenir en un mòdul concret. Per exemple, en el cas del login, s'ha tingut en compte si el correu que s'ha enviat existeix en la bbdd, per enviar una resposta, si la contrasenya és la mateixa, o no, etc.. I d'aquesta manera podem tenir una resposta més personalitzada.

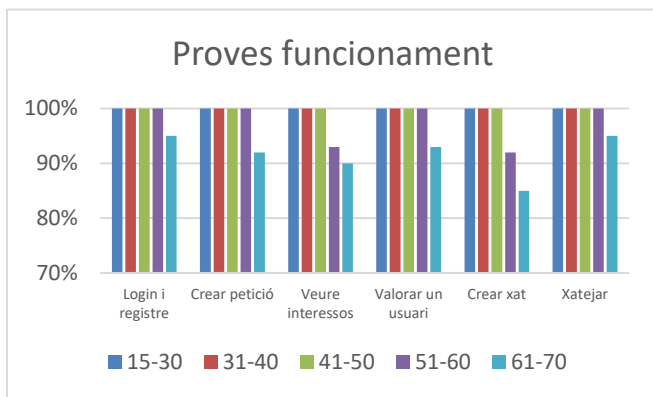
Per tant, en el backend s'han realitzat dos tipus de proves principalment, que són de caixa blanca i caixa negra. Les proves de caixa negra, consisteix en introduir un input a través de Postman i veure si la resposta és l'esperada. I l'altra forma de testeig, enfocat al frontend, és realitzar comprovacions per respondre al frontend i així poder tenir errors més personalitzats a l'hora de la gestió d'errors dins del frontend.

10 RESULTATS

La obtenció de resultats ha estat un punt clau del projecte ja que ha sigut una base per saber que els objectius s'estaven complint i que l'experiència d'usuari estava sent la esperada. Aquests resultats s'han pogut realitzar a través de la validació dels testos, i a més s'ha donat la possibilitat a diferents usuaris de diferents edats, el poder provar l'aplicació per poder demostrar aquesta fiabilitat i que la web és intuïtiva.

Aquesta prova s'ha realitzat fent un test a un total de 18 persones, les quals s'establien entre les edats de 15-30 (5 persones), 31-40 (4 persones), 41-50 (3 persones), 51-60 (3 persones), 61-70 (3 persones). Les proves han consistit en demanar als voluntaris a realitzar alguna acció concreta, com per exemple, registrar-se i iniciar sessió, crear una petició de canguratge, veure els interessos, valorar un usuari, crear un xat, xatejar, etc.

Per tant, a continuació es mostrarà les proves realitzades amb els resultats obtinguts.



Els resultats de les proves ha sigut molt favorable, ja que en la franja d'edat de 15 a 50 el 100% dels participants han tingut un acert del 100%, el que vol dir que l'app és intuïtiva. Tot i això, les franges d'edat superior a 50, també han tingut percentatges superiors al 90%. El que vol dir que l'objectiu de fer una web intuïtiva s'ha complert, ja que les persones de major edat, com per exemple, els avis, també poden utilitzar la web per poder trobar un canguratge.

Com a conclusió dels resultats, s'ha demostrat a través de proves i de la validació dels testos que el funcionament de la web és l'esperat, i que a més la web pot ser utilitzada per qualsevol franja d'edat fins als 70 anys com a mínim.

11 CONCLUSIONS

El desenvolupament d'aquest projecte ha estat motivat per dos aspectes destacats.

En primer lloc, aquest projecte ha estat un repte a nivell professional, ja que en el punt inicial del projecte no tenia cap coneixement sobre les tecnologies utilitzades en el projecte, excepte oracle. Per tant, durant tot el projecte he hagut d'anar formant-me sobre diferents aspectes del projecte, connexions, forma d'estructurar els mòduls, la particularitat dels components de reactjs que és una forma de treballar molt diferent a la d'altres llenguatges, sobre tot a l'hora de treballar amb els props, connexió amb l'API, etc. Tot i això, el projecte ha servit per poder tenir una base sobre aquestes tecnologies i per ampliar els coneixements.

A més, el projecte ha estat molt satisfactori a nivell personal, ja que m'agrada molt treballar amb nens, i la motivació del projecte és gràcies a haver buscat en altres pàgines sobre canguratge i no haver estat satisfet amb l'experiència.

A nivell global, aquest projecte és un projecte innovador, amb molt marge de millora i d'ampliació, però que té una versió estable i validada que pot assegurar una persistència i un bon funcionament.

Per tant, per concloure aquest projecte, podem dir que s'han assolit tots els objectius principals, i una part dels objectius secundaris i finalment, s'ha desenvolupat una web

intuïtiva, fàcil d'utilitzar, la qual permet a l'usuari escollir la millor opció per cuidar al seu fill, sense necessitat de sortir de la web. A més també s'ha desenvolupat una web on es pot crear usuaris, crear peticions d'usuaris, valorar usuaris, veure les valoracions d'un usuari, crear, modificar i eliminar peticions de canguratge, interessar-se per peticions de canguratge, crear xats amb usuaris i xatejar amb usuaris.

12 AGRAÏMENTS

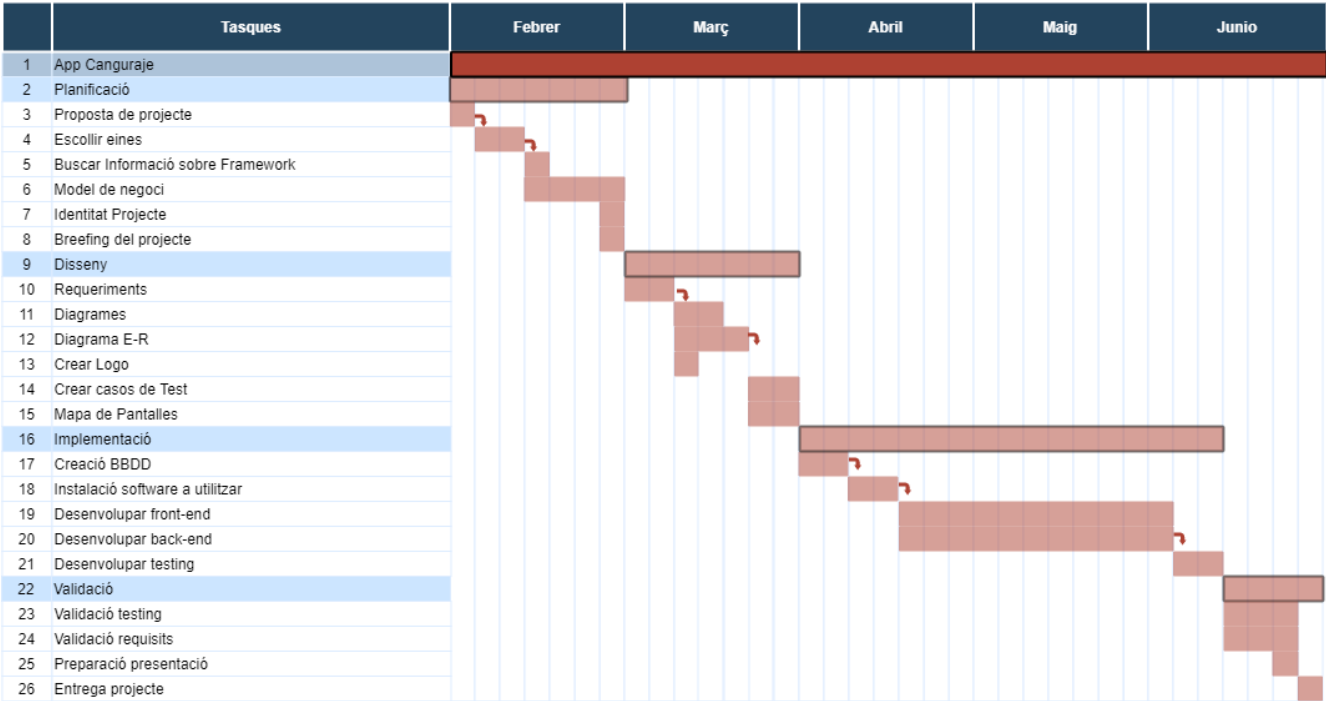
L'autor vol agrair a totes les persones que han dedicat un moment del seu temps per fer la prova i poder així obtenir uns resultats més complets, i per suposat, al tutor del TFG, Carles Sanchez per aconsellar i ajudar en tot el possible al desenvolupament del projecte i el seguiment del mateix.

BIBLIOGRAFIA

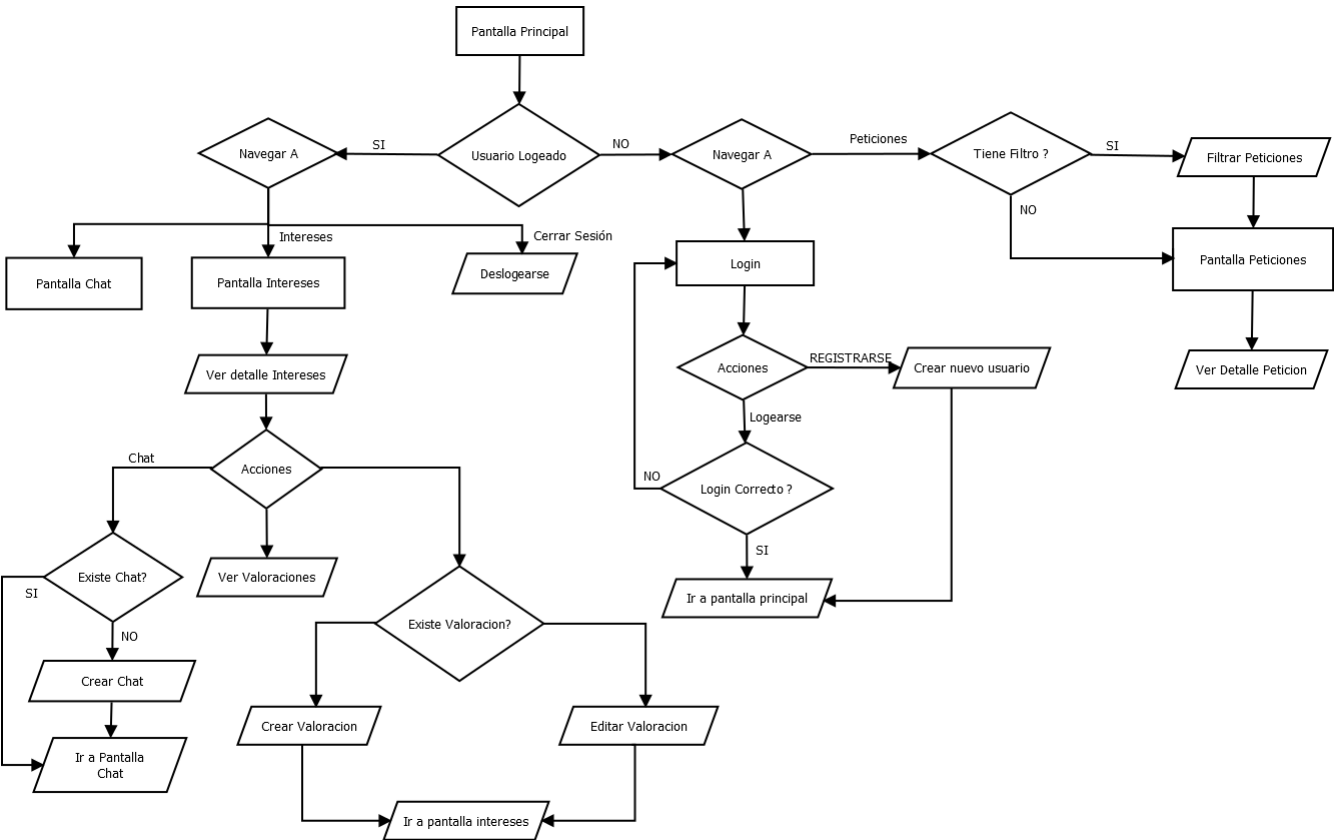
- [1] <http://educacio.gencat.cat/ca/actualitat/escolasegura/tracacovid/dades-covid19-centres/#/>
- [2] <https://elpais.com/tecnologia/2020-03-11/la-suspension-de-clases-dispara-los-anuncios-online-para-cuidar-ninos-hasta-niveles-nunca-vistos.html>
- [3] <https://www.waremarketing.com/es/blog/frameworks-en-el-desarrollo-web-las-mejores-practicas-para-tu-negocio-online.html>
- [4] <https://www.tithink.com/es/2018/11/14/7-razones-para-utilizar-react/#:~:text=Como%20ya%20hemos%20mencionado%20anteriormente,la%20comunicaci%C3%B3n%20con%20los%20dem%C3%A1s>
- [5] <https://openwebinars.net/blog/frameworks-para-frontend-mas-populares-en-2020/>
- [6] <http://cangursdeguardia.com/>
- [7] https://www.sitly.es/?gclid=Cj0KCQjwse-DBhC7ARIsAI8YcWKR8cfmUkuec0rbYDkYMMB12oYrJvxDe1i_USmoOq6F7F7OpHbr1RoaAjtyEALw_wcB
- [8] <https://www.milanuncios.com/>
- [9] <https://www.linkedin.com/>
- [10] <https://www.infojobs.net/>
- [11] <https://angular.io/>
- [12] <https://vuejs.org/>
- [13] <https://es.reactjs.org/>
- [14] <https://www.arsys.es/blog/programacion/axios/>
- [15] <https://es.reactjs.org/docs/render-props.html#:~:text=El%20t%C3%A9rmino%20E2%80%9Crender%20prop%20E2%80%9D%20se,so%20propia%20l%C3%B3gica%20de%20representaci%C3%B3n>
- [16] <https://midu.dev/react-hooks-introduccion-saca-todo-el-potencial-sin-class/#:~:text=Los%20Hooks%20son%20una%20nueva,componentes%20creados%20con%20una%20funcion%20.&text=Hooks%20es%20gancho%20y%20C%20precisamente,las%20caracter%C3%ADsticas%20que%20ofrece%20React>
- [17] <https://enekodelatorre.com/expressjs-instalacion-primeros-pasos/>
- [18] <https://www.oracle.com/es/database/technologies/appdev/datamodeler.html>
- [19] <https://www.oracle.com/database/technologies/appdev/sql-developer-landing.html>
- [20] <https://codingpotions.com/react-componentes>

APÈNDIX

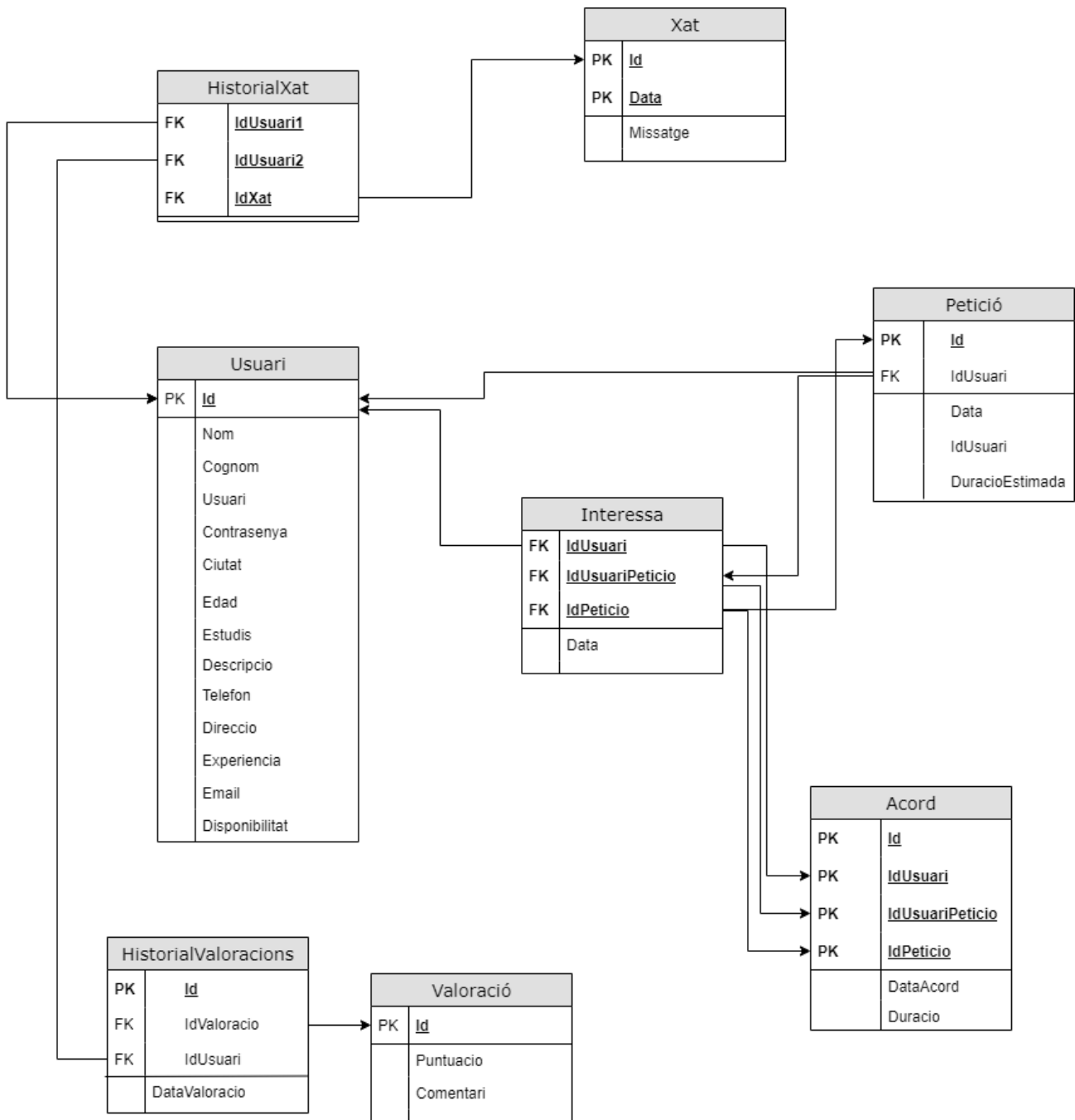
A1. DIAGRAMA DE GANTT



A2. DIAGRAMA DE FLUX



A3. DIAGRAMA MODELER BASE DE DADES



A4. diagrama E-R base de dades

